

3

### Abstract of the Disclosure

A generic fault tolerant "match-and-set" locking mechanism and method for controlling access to resources shared among a plurality of users  $N$ . The match-and-set lock has a locked operating state and an unlocked operating state controlled by the value  $C$  of its content such that the lock is in its locked operating state when  $C \neq 0$  and in its unlocked operating state where  $C=0$ . The lock returns a value  $R$ , equal to the lock's current content  $C$ , to an inquiring user seeking access to the resource. A return value  $R=0$  denotes that the resource is free, and a return value  $R \neq 0$  denotes that the resource is locked by another user. The lock is responsive to a command in the form  $(A, B)$ , such that the lock substitutes  $B$  for  $C$  if  $A=C$ . Thus, the lock may be locked by issuing the command  $(A, B)$  where  $A=C$  and  $B \neq 0$ ; and the lock may be released by issuing the command  $(A, B)$  where  $A=C$  and  $B=0$ . A deadlock condition may be avoided by setting the lock to the value  $B=P+T*(N+1)$ , where  $P < (N+1)$  and identifies the user issuing this command  $(A, B)$ , and  $T$  is the current global time stamp when the user issues this command. When the return value  $R \neq 0$ ,  $R$  identifies the user currently locking the resource and the time when that user locked the resource: the locking user id is simply  $R\%(N+1)$ , and the locking time is  $R/(N+1)$ . An inquiring user can then determine if the user currently locking the resource has failed, or restarted, since locking the resource. If so, the inquiring user can rescue the lock from the control of the failed user. The mechanism of "match-and-set" ensures that exactly one such user can succeed in rescuing that lock. Crashed users can come back online and participate in new locking operations: the locking semantics is always maintained and this system suffers no deadlocks. The implementation requirements are easy to satisfy: that there is a global reference clock, that any user maintains a time stamp (off the global clock) of its latest reboot time, and the match-and-set access is an indivisible operation. The last can be implemented in hardware in a way similar to that of the traditional test-and-set lock. The plurality of users may be processors in a multiprocessor computer system, and the processors may form a telecommunications system.